Homework Assignment Number Ten Assignment Solutions

Problem 1.
Find the solution to the following system of nonlinear algebraic equations near (1,1,1).

$$f_1 = x_1 + 2x_2 + 3x_3 - 4$$
$$f_2 = {x_1}^3 - 4{x_2}^3$$
$$f_3 = x_3 - sin\, x_3$$

Solution:

Since this is a system of three nonlinear algebraic equations with three unknowns, we can use the multivariate Newton-Raphson with numerical approximations to the derivatives. We will set a tolerance of 10^{-6}.
The input function looks like:

```
function f = funkeval(x)
%
%  these two lines force a column vector of length n
%
n = max(size(x));
f = zeros(n,1);
%
%  enter the functions here
%
f(1) = x(1) + 2*x(2) + 3*x(3) - 4;
f(2) = x(1)^3 - 4*x(2)^3;
f(3) = x(3) - sin(x(3));
```

The command line prompt and output yields

```
>> [x,err,f] = nrndn([1,1,1],1.0e-6,1)
iter =    1, err =  3.28e-01 f =  2.08e+00
 iter =    2, err =  3.97e-01 f =  5.13e-01
 iter =    3, err =  1.50e-01 f =  6.63e-01
...
 iter =   31, err =  1.81e-06 f =  2.95e-10
 iter =   32, err =  1.21e-06 f =  1.97e-10
 iter =   33, err =  8.03e-07 f =  1.31e-10

x =   1.769971344010968   1.115012077004957   0.000001500659706

err =     8.031984760117370e-07

f =     1.310558041829265e-10
```

Therefore, the solution is $x_1 = 1.76997, x_2 = 1.11501 \text{and} x_3 = 0$.

Problem 2.
Perform 2 iterations of multivariate Newton-Raphson on the following system of nonlinear algebraic equations.

$$f_1 = {x_1}^2 + {x_2}^2 - 4$$
$$f_2 = {x_1}^2 - x_2 + 1$$

Show the value of the Jacobian, residual and $\underline{\delta x}^{(k)}$ at each iteration. Use and initial guess of (x_1, x_2) = (1,2). Report the values of (x_1, x_2) for the first two iterations.

Solution:

In order to perform the multivariate Newton-Raphson method, we must first determine the functional form of the partial derivatives

$$\begin{array}{ll} \left(\underline{J}\right)_{1,1} = \left(\frac{\partial f_1}{\partial x_1}\right) = 2x_1 & \left(\underline{J}\right)_{1,2} = \left(\frac{\partial f_1}{\partial x_2}\right) = 2x_2 \\ \left(\underline{J}\right)_{1,2} = \left(\frac{\partial f_2}{\partial x_1}\right) = 2x_1 & \left(\underline{J}\right)_{2,2} = \left(\frac{\partial f_2}{\partial x_2}\right) = -1 \end{array} \rightleftarrows$$

Then following the algorithm outlined above:
Step One. Make an initial guess. In this case, the initial guess is given in the problem statement: x_1= 1.0 and x_2 = 2.0 .
Step Two. Using that initial guess, calculate the residual and the Jacobian.

$$\underline{J}^{(0)} = \begin{bmatrix} 2 & 4 \\ 2 & -1 \end{bmatrix} \text{ and } \underline{R}^{(0)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Step Three. Solve $\underline{J}^{(k)}\underline{\delta x}^{(k)} \rightleftarrows = -\underline{R}^{(k)}$ (Using Linear Algebra)

$$\underline{\delta x}^{(0)} = \begin{bmatrix} -0.1 \\ -0.2 \end{bmatrix}$$

Step Four. Calculate new values for $\underline{x}$

$$\underline{x}^{(1)} = \underline{x}^{(0)} + \underline{\delta x}^{(0)} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} + \begin{bmatrix} -0.1 \\ -0.2 \end{bmatrix} = \begin{bmatrix} 0.9 \\ 1.8 \end{bmatrix}$$

Step Five. Loop back to Step 2. and repeat until converged.

Here are what further iterations yield

iteration	$\underline{x}$	$\underline{J}$	$\underline{R}$	$\underline{\delta x}$
0	$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$	$\begin{bmatrix} 2 & 4 \\ 2 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} -0.1 \\ -0.2 \end{bmatrix}$
1	$\begin{bmatrix} 0.9 \\ 1.8 \end{bmatrix}$	$\begin{bmatrix} 1.8 & 3.6 \\ 1.8 & -1 \end{bmatrix}$	$\begin{bmatrix} 0.05 \\ 0.01 \end{bmatrix}$	$\begin{bmatrix} -0.0104 \\ -0.0087 \end{bmatrix}$
2	$\begin{bmatrix} 0.8896 \\ 1.7913 \end{bmatrix}$	$\begin{bmatrix} 1.7792 & 3.5826 \\ 1.7792 & -1.0000 \end{bmatrix}$	$\begin{bmatrix} 0.1835e-3 \\ 0.1079e-3 \end{bmatrix}$	$\begin{bmatrix} \text{-0.6991e-4} \\ \text{-0.1650e-4} \end{bmatrix}$

Problem 3.
Consider the solid-liquid equilibrium of a binary alloy described by regular solution theory. Consider two materials, A and B with the following pure component properties:

component	melting temperature (K)	enthalpy of melting (J/mol)	entropy of melting (J/mol/K)
symbol	T_A^{melt}, T_B^{melt}	$\Delta H_A^{melt}, \Delta H_B^{melt}$	$\Delta S_A^{melt}, \Delta S_B^{melt}$
A	800.0	8,000.0	10.0
B	1200.0	12,000.0	10.0

The regular solution parameters for the enthalpy of mixing are given as

	liquid	solid
enthalpy of mixing parameter (J/mol) (Ω)	-5,000.0	0.0

(a) Plot the free energy curve of each phase at a temperature of T = 1000 K as a function of composition.
(b) Determine the liquid and solid equilibrium compositions at T = 1000 K.

Regular solution theory in a nutshell.
The total free energy of either liquid (liq) or solid (sol) phase is made up of two terms, one representing a phase change if the phase is not the equilibrium state and one representing the contribution due to mixing.

$$\Delta G_{liq}^{\square} = \Delta G_{liq}^{phase} + \Delta G_{liq}^{mix} \qquad (1.liq)$$

$$\Delta G_{sol}^{\square} = \Delta G_{sol}^{phase} + \Delta G_{sol}^{mix} \qquad (1.sol)$$

The mixing term has the following form which is the sum of an enthalpy term and an entropy term.

$$\Delta G_{liq}^{mix} = \Omega_{liq} x_A x_B + RT[x_A ln(x_A) + x_B ln(x_B)] \qquad (2.liq)$$

$$\Delta G_{sol}^{mix} = \Omega_{sol} x_A x_B + RT[x_A ln(x_A) + x_B ln(x_B)] \qquad (2.sol)$$

where R is the gas constant (8.314 J/mol/K) and where x_A is the atomic fraction of A, and x_B is the atomic fraction of B. Remember, in a binary mixture, x_B is not an independent variable. Rather, $x_B = 1 - x_A$. Note: Because of the natural logarithm in equation (2), the free energy does not exist exactly at $x_A = 0$ or $x_A = 1$.

The phase change term has the following form.

$$\Delta G_{liq}^{phase} = x_A \Delta G_{liq,A}^{phase} + x_B \Delta G_{liq,B}^{phase} \qquad (3.liq)$$

$$\Delta G_{sol}^{phase} = x_A \Delta G_{sol,A}^{phase} + x_B \Delta G_{sol,B}^{phase} \qquad (3.sol)$$

where $dG_{liq,A}^{phase}$ and $dG_{sol,A}^{phase}$ are phase change free energies of pure component A in the liquid and solid phases respectively. They are defined as

$$\Delta G_{liq,A}^{phase} = \begin{cases} \Delta G_A^{melt} & if\ T < T_A^{melt} \\ 0 & if\ T \geq T_A^{melt} \end{cases} \qquad (4.liq)$$

$$\Delta G_{sol,A}^{phase} = \begin{cases} 0 & if\ T \leq T_A^{melt} \\ -\Delta G_A^{melt} & if\ T > T_A^{melt} \end{cases} \qquad (4.sol)$$

where

$$\Delta G_A^{melt} = \Delta H_A^{melt} - T_A^{melt} \Delta S_A^{melt} \tag{5}$$

Equations (4) and (5) were written for component A. Analogous versions can be written for component B.

For part (b) of the problem, we will solve two equations for two unknowns. The two equations can be understood from the point of view of the common tangent construction. There is a tangent line to the liquid free energy curve and a second tangent line to solid free energy curve, given in equation (a). These tangent lines have the age-old form.

$$y = mx + b \tag{6}$$

where m is the slope of the line and b is the y-intercept. At the equilibrium liquid and solid composition we can write this equation as

$$y_{liq} = m_{liq} x_{liq,A} + b_{liq} \tag{7.liq}$$

$$y_{sol} = m_{sol} x_{sol,A} + b_{sol} \tag{7.sol}$$

To be clear, our goal is to find the unknown liquid and solid compositions, $x_{liq,A}$ and $x_{sol,A}$. Because there is a common tangent, we have two equations that define the unknown compositions, namely the slopes are the same and the y-intercepts are the same.

$$f_1\left(x_{liq,A}, x_{sol,A}\right) = m_{liq} - m_{sol} = 0 \tag{8.i}$$

$$f_2\left(x_{liq,A}, x_{sol,A}\right) = b_{liq} - b_{sol} = 0 \tag{8.ii}$$

To evaluate equation (8.i), we must recognize that the slope of the liquid or solid free energy curve is the partial derivative of that curve with respect to the atomic fraction of A.

$$m_{liq} = \left. \frac{\partial \Delta G_{liq}^{\square}}{\partial x_A} \right|_{x_{liq,A}} \tag{9.liq}$$

$$m_{sol} = \left. \frac{\partial \Delta G_{sol}^{\square}}{\partial x_A} \right|_{x_{sol,A}} \tag{9.sol}$$

We can rearrange equation (7) to solve for the intercepts

$$b_{liq} = y_{liq} - m_{liq} x_{liq,A} \tag{10.liq}$$

$$b_{sol} = y_{sol} - m_{sol} x_{sol,A} \tag{10.sol}$$

where y_{liq} and y_{sol} are points on the respective free energy curves (equation 1) evaluated at $x_{liq,A}$ and $x_{sol,A}$.

$$y_{liq} = \Delta G_{liq}^{\square}\left(x_{liq,A}\right) \tag{11.liq}$$

$$y_{sol} = \Delta G_{sol}^{\square}\left(x_{sol,A}\right) \tag{11.sol}$$

Substitution of equations (9), (10) and (11) into equation (8) yields

$$f_1\left(x_{liq,A}, x_{sol,A}\right) = \left.\frac{\partial \Delta G_{liq}^{\square}}{\partial x_A}\right|_{x_{liq,A}} - \left.\frac{\partial \Delta G_{sol}^{\square}}{\partial x_A}\right|_{x_{sol,A}} = 0 \tag{12.i}$$

$$f_2\left(x_{liq,A}, x_{sol,A}\right) = \Delta G_{liq}^{\square}\left(x_{liq,A}\right) - \left.\frac{\partial \Delta G_{liq}^{\square}}{\partial x_A}\right|_{x_{liq,A}} x_{liq,A} - \Delta G_{sol}^{\square}\left(x_{sol,A}\right) + \left.\frac{\partial \Delta G_{sol}^{\square}}{\partial x_A}\right|_{x_{sol,A}} x_{sol,A} = 0 \tag{12.ii}$$

Addendum: Help with obtaining the analytical derivative of free energy with respect to the atomic fraction of A.

Start by differentiating equation (1)

$$\frac{\partial \Delta G_{liq}^{\square}}{\partial x_A} = \frac{\partial \Delta G_{liq}^{phase}}{\partial x_A} + \frac{\partial \Delta G_{liq}^{mix}}{\partial x_A} \tag{13.liq}$$

$$\frac{\partial \Delta G_{sol}^{\square}}{\partial x_A} = \frac{\partial \Delta G_{sol}^{phase}}{\partial x_A} + \frac{\partial \Delta G_{sol}^{mix}}{\partial x_A} \tag{13.sol}$$

Next we differentiate the mixing term in equation (2). Remember, in a binary mixture, x_B is not an independent variable. Rather, $x_B = 1 - x_A$. So, first rewrite equation (2) substituting $1 - x_A$ for x_B.

$$\Delta G_{liq}^{mix} = \Omega_{liq} x_A(1 - x_A) + RT[x_A ln(x_A) + (1 - x_A) ln(1 - x_A)] \tag{14.liq}$$

$$\Delta G_{sol}^{mix} = \Omega_{sol} x_A(1 - x_A) + RT[x_A ln(x_A) + (1 - x_A) ln(1 - x_A)] \tag{14.sol}$$

Now differentiate with respect to x_A. We left out intermediate steps. Check for yourself.

$$\frac{\partial \Delta G_{liq}^{mix}}{\partial x_A} = \Omega_{liq}(1 - 2x_A) + RTln\left(\frac{x_A}{1 - x_A}\right) \tag{15.liq}$$

$$\frac{\partial \Delta G_{sol}^{mix}}{\partial x_A} = \Omega_{sol}(1 - 2x_A) + RTln\left(\frac{x_A}{1 - x_A}\right) \tag{15.sol}$$

Next we differentiate the phase change term in equation (3). Remember, in a binary mixture, x_B is not an independent variable. Rather, $x_B = 1 - x_A$. So, first rewrite equation (3) substituting $1 - x_A$ for x_B.

$$\Delta G_{liq}^{phase} = x_A \Delta G_{liq,A}^{phase} + (1 - x_A)\Delta G_{liq,B}^{phase} \tag{16.liq}$$

$$\Delta G_{sol}^{phase} = x_A \Delta G_{sol,A}^{phase} + (1 - x_A)\Delta G_{sol,B}^{phase} \tag{16.sol}$$

Now differentiate with respect to x_A.

$$\frac{\partial \Delta G_{liq}^{phase}}{\partial x_A} = \Delta G_{liq,A}^{phase} - \Delta G_{liq,B}^{phase} \tag{17.liq}$$

$$\frac{\partial \Delta G_{sol}^{phase}}{\partial x_A} = \Delta G_{sol,A}^{phase} - \Delta G_{sol,B}^{phase} \tag{17.sol}$$

We now have everything we need to solve this problem.

Solution:

First, I created a script called driver.m to call functions to perform both of the tasks for part (a) and part (b).

```
clear all;
close all;
%
%  make free energy plot
%
make_freeenergy_plot;

%
% solve for equilibrium compositions
%
%
% make initial guess
%
xsol = 0.10;
xliq = 0.30;
x0 = [xsol,xliq];
fprintf(1,'initial guess xsol = %f & xliq = %f\n\n',x0);
%
%  call Newton-Raphson
%
tol = 1.0e-6;
iprint = 1;
[x,err,f] = nrndn(x0,tol,iprint);
fprintf(1,'\nconverged values xsol = %f & xliq = %f\n',x);
fprintf(1,'\nRMS error at converged solution = %e \n',err);
```

Second, I created a function called make_freeenergy_plot.m to make the liquid and solid free energy curvers as a function of composition.

```
function make_freeenergy_plot
% parameters
R = 8.314; % J/mol/K
omega_sol =  0.0; % J/mol
omega_liq =  -5000.0; % J/mol
TA = 800.0; % K
TB = 1200.0; % K
dHmeltA = 8000.0; % J/mol
dHmeltB = 12000.0; % J/mol
dSmeltA = 10.0; % J/mol/K
dSmeltB = 10.0; % J/mol/K
%
%  temperature
%
T = 1000.0; % K

%
%  create vector of mole fractions of A
%  avoid 0.0 and 1.0 due to log
%
xAvec = [0:0.01:1];
nxA = max(size(xAvec));
xAvec(1) = 1.0e-3;
xAvec(nxA) = 1.0 - 1.0e-3;
%
dGliq = zeros(1,nxA);
dGsol = zeros(1,nxA);
%
%   define phase change values
%
dGmeltA = dHmeltA - T*dSmeltA;
dGmeltB = dHmeltB - T*dSmeltB;
if (T <= TA && T <= TB)
    dGphase_liq_A = dGmeltA;
    dGphase_liq_B = dGmeltB;
    dGphase_sol_A = 0;
    dGphase_sol_B = 0;
elseif (T >= TA && T >= TB)
    dGphase_sol_A = -dGmeltA;
    dGphase_sol_B = -dGmeltB;
    dGphase_liq_A = 0;
    dGphase_liq_B = 0;
elseif (T > TA && T < TB)
    dGphase_sol_A = -dGmeltA;
    dGphase_sol_B = 0;
    dGphase_liq_A = 0;
    dGphase_liq_B = dGmeltB;
end
%
% loop over mole fractions
%
for ix = 1:1:nxA
   xA = xAvec(ix);
   dGphase_liq = xA*dGphase_liq_A + (1-xA)*dGphase_liq_B;
   dGphase_sol = xA*dGphase_sol_A + (1-xA)*dGphase_sol_B;
   term2 = R*T*(xA*log(xA) + (1-xA)*log(1-xA));
   dGmix_sol =  omega_sol*xA*(1-xA)  + term2;
   dGmix_liq =  omega_liq*xA*(1-xA)  + term2;
   dGliq(ix) = dGphase_liq + dGmix_liq;
   dGsol(ix) = dGphase_sol + dGmix_sol;
end
figure(1)
plot(xAvec,dGliq,'k')
hold on;
plot(xAvec,dGsol,'r')
hold off;
legend('liquid','solid')
title(strcat('T = ',int2str(T),' K'))
xlabel('mole fraction of A');
ylabel('Gibbs free energy change (J/mol)');
```

Third, I modified the funkeval function associated with the Newton Raphson with Numerical Derivatives for N equations code (nrndn.m) as follows:

```
function f = funkeval(x)
%
%  regular solution parameters
%
R = 8.314; % J/mol/K
omega_sol =  0.0; % J/mol
omega_liq =  -5000.0; % J/mol
TA = 800.0; % K
TB = 1200.0; % K
dHmeltA = 8000.0; % J/mol
dHmeltB = 12000.0; % J/mol
dSmeltA = 10.0; % J/mol/K
dSmeltB = 10.0; % J/mol/K
%
%  temperature
%
T = 1000.0; % K
%
%  these two lines force a column vector of length n
%
n = max(size(x));
f = zeros(n,1);
%
%  identify variables
%
xA_sol = x(1);
xA_liq = x(2);
%
%   define phase change values
%
dGmeltA = dHmeltA - T*dSmeltA;
dGmeltB = dHmeltB - T*dSmeltB;
```

```
if (T <= TA && T <= TB)
        dGphase_liq_A = dGmeltA;
        dGphase_liq_B = dGmeltB;
        dGphase_sol_A = 0;
        dGphase_sol_B = 0;
elseif (T >= TA && T >= TB)
        dGphase_sol_A = -dGmeltA;
        dGphase_sol_B = -dGmeltB;
        dGphase_liq_A = 0;
        dGphase_liq_B = 0;
elseif (T > TA && T < TB)
        dGphase_sol_A = -dGmeltA;
        dGphase_sol_B = 0;
        dGphase_liq_A = 0;
        dGphase_liq_B = dGmeltB;
end
%
%  phase change component of free energy
%
dGphase_liq = xA_liq*dGphase_liq_A + (1-xA_liq)*dGphase_liq_B;
dGphase_sol = xA_sol*dGphase_sol_A + (1-xA_sol)*dGphase_sol_B;
%
% mixing component of free energy
%
term2_liq = R*T*(xA_liq*log(xA_liq) + (1-xA_liq)*log(1-xA_liq));
term2_sol = R*T*(xA_sol*log(xA_sol) + (1-xA_sol)*log(1-xA_sol));
dGmix_sol =  omega_sol*xA_sol*(1-xA_sol) + term2_sol;
dGmix_liq =  omega_liq*xA_liq*(1-xA_liq) + term2_liq;
%
%  total free energy
%
dG_liq = dGphase_liq + dGmix_liq;
dG_sol = dGphase_sol + dGmix_sol;
%
%  the partial derivative of the free energy
%
dGdxA_phase_liq =  dGphase_liq_A - dGphase_liq_B;
dGdxA_phase_sol =  dGphase_sol_A - dGphase_sol_B;
dterm2dxA_liq = R*T*(log(xA_liq/(1-xA_liq)));
dterm2dxA_sol = R*T*(log(xA_sol/(1-xA_sol)));
dGdxA_mix_liq = omega_liq*(1-2*xA_liq) + dterm2dxA_liq;
dGdxA_mix_sol = omega_sol*(1-2*xA_sol) + dterm2dxA_sol;
dGdxA_liq = dGdxA_phase_liq + dGdxA_mix_liq;
dGdxA_sol = dGdxA_phase_sol + dGdxA_mix_sol;
%
%  the equations
%
% slopes are equal
f(1) = dGdxA_liq - dGdxA_sol;
% intercepts are equal
intercept_liq = dG_liq - dGdxA_liq*xA_liq;
intercept_sol = dG_sol - dGdxA_sol*xA_sol;
f(2) = intercept_liq - intercept_sol;
```

I executed the script, driver.m, and obtained the following output.

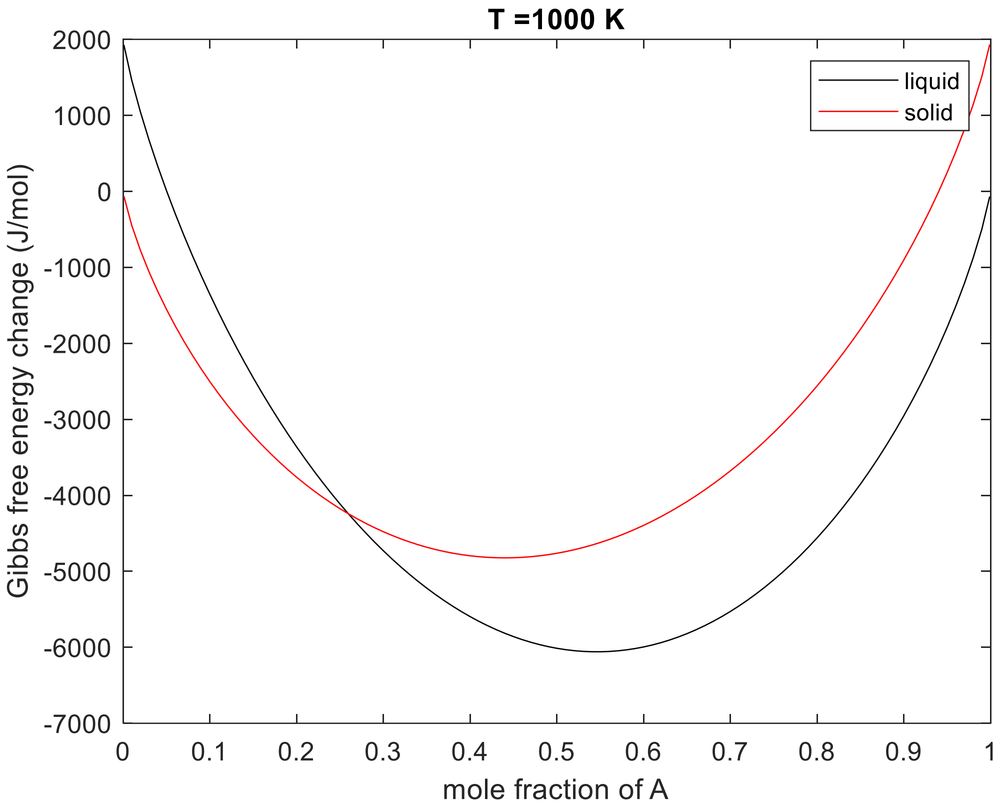
```
>> driver
initial guess xsol = 0.100000 & xliq = 0.300000

iter =    1, err =  3.93e-02 f =  3.71e+03
iter =    2, err =  2.81e-02 f =  6.58e+02
iter =    3, err =  5.00e-03 f =  9.57e+01
iter =    4, err =  1.34e-04 f =  2.36e+00
iter =    5, err =  8.94e-08 f =  1.58e-03

converged values xsol = 0.194273 & xliq = 0.325002
```

RMS error at converged solution = 8.937836e-08

and the plot



Therefore, the solution is $x_{liq,A} = 0.194$, $x_{sol,A} = 0.325$.

Note: If you have trouble getting the Newton Raphson method to converge, you can extract reasonably good guesses for the solution from the plot in part (a) as initial guesses for part (b).